Dynamic Objects Effect on Visibility Analysis in 3D Urban Environments

Oren Gal and Yerach Doytsher

Mapping and Geo-information Engineering
Technion - Israel Institute of Technology
Haifa, Israel
{orengal,doytsher}@technion.ac.il

Abstract. This paper presents a unique formulation and concept of the dynamic object effect on constant objects, such as buildings, dealing with visibility problems in 3D urban environments. Dynamic objects in a 3D urban environment, such as cars, pedestrians and trees, are usually omitted from visibility analysis. In order to challenge this problem, we focus on modeling predicting and estimating dynamic objects' future location in the environment. We integrate all these factors into our visibility analysis and create a probabilistic visibility analysis, changed over time due to the dynamic character of these objects.

Our probabilistic visibility concept takes into account 3D boxes and cylinders, generating a fast and exact analytic solution to dynamic and static objects; to illustrate our concept, we use web-cameras located at constant points in the treated environments in order to update our model in each time period from web source data and to analyze the environment. Dynamic objects prediction is based on validated models for driver behavior; pedestrians' walking routes, trees' displacement and wind effect. A real urban environment with dynamic objects approximated by 3D boxes and cylinders demonstrates our approach.

Keywords: Visibility, Spatial Analysis, Urban Environments, 3D.

1 Introduction and Related Work

The visibility problem has been extensively studied over the last twenty years, due to the importance of visibility in GIS and Geomatics, computer graphics and computer vision, and robotics. Visibility analysis in open terrains has been extensively studied, mostly by using already-known Digital Terrain Models (DTM). Only a few research projects have treated visibility analysis in 3D dense environments such as urban scenes, and as of now, have still not been targeted as a major research field.

Accurate visibility computation in 3D environments is a very complicated task demanding a high computational effort, which can hardly be carried out in a very short time using traditional well-known visibility methods [15]. The exact visibility methods are highly complex, and cannot be used for fast applications due to their long computation time. As mentioned above, previous research in visibility computation

S. Liang, X. Wang, and C. Claramunt (Eds.): W2GIS 2013, LNCS 7820, pp. 147–163, 2013.

has been devoted to open environments using DEM models, representing raster data in 2.5D (Polyhedral model), and do not address, or suggest solutions for, dense built-up areas. Most of these published papers have focused on approximate visibility computation, enabling fast results using interpolations of visibility values between points, calculating point visibility with the Line of Sight (LOS) method [2]. Other fast algorithms are based on the conservative Potentially Visible Set (PVS) [3]. These methods are not always completely accurate, as they may render hidden objects' parts as visible due to various simplifications and heuristics.

A vast number of algorithms have been suggested for speeding up the process and reducing computation time. Franklin [7] evaluates and approximates visibility for each cell in a DEM model based on greedy algorithms. Wang et al. [22] introduced a Grid-based DEM method using viewshed horizon, saving computation time based on relations between surfaces and the line of sight (LOS) method. Later on, an extended method for viewshed computation was presented, using reference planes rather than sightlines [23].

One of the most efficient methods for DEM visibility computation is based on shadow-casting routine. The method casts shadowed volumes in the DEM, like a light bubble [17]. Extensive research has treated Digital Terrain Models (DTM) in open terrains, mainly Triangulated Irregular Network (TIN) and Regular Square Grid (RSG) structures. Visibility analysis in terrain was classified into point, line and region visibility, and several algorithms have been introduced, based on horizon computation describing a visibility boundary [5].

Only a few works have treated visibility analysis in urban environments. A mathematical model of an urban scene, calculating probabilistic visibility for a given object from a specific viewcell in the scene, has been presented by [11]. This is a very interesting concept, which extends the traditional deterministic visibility concept. Nevertheless, the buildings are modeled as cylinders, and the main challenges of spatial analysis and building model were not tackled. Other methods have been developed, subject to computer graphics and vision fields, dealing with exact visibility in 3D scenes, without considering environmental constraints. Plantinga and Dyer [15] used the aspect graph – a graph with all the different views of an object.

Lately, several spatial analysis methods for urban environments have been presented, using constant known 3D models tailor-made for spatial aspects [24]. An urban environment can be divided into static objects such as buildings, roads, etc., and dynamic objects such as moving cars, tree branches, pedestrians etc.

Static objects which are modeled by 3D GIS models are not frequently changed. On the other hand, dynamic objects cannot be modeled efficiently by a constant 3D model, and must be updated from on-line sources, such as web cameras, in a certain time period.

Visibility analysis from point A to B is mostly captured as Boolean values of 'visible' and 'invisible' ("1" and "0" values). The complex modeling of these dynamic objects, and the uncertainties during the time update gap using on-line data sources, such as web cameras, in the environments, to update the urban environment model, lead to the probabilistic visibility analysis concept. This concept sets probability value for object visibility.

In this paper we present the probabilistic visibility concept in urban environments, taking into account dynamic objects and their effect on visibility analysis of constant objects such as buildings. In order to address this problem, we focus on modeling, predicting and estimating their future location in the environment. We integrate all these factors into our visibility analysis and create a probabilistic visibility analysis which changes over time, due to the dynamic character of these objects.

Our probabilistic visibility concept is related only to dynamic objects; static objects, such as buildings, are analyzed by using an efficient visibility analysis method presented by [8].

To illustrate our concept, we use web-cameras located at constant points in the environments in order to update our model in each time period (as is already being done in Google Maps application).

We start by mapping dynamic objects in an urban environment, modeling them for efficient visibility analysis, and defining a probabilistic visibility model for each of them.

2 Dynamic Objects – Modeling and Probabilistic Visibility

Dynamic objects such as moving cars, tree branches, pedestrians etc., directly affect visibility in urban environments.

Due to modeling limitations, these entities are usually neglected in spatial analysis aspects. We focus on three major dynamic objects in an urban case: moving cars, tree branches and pedestrians. Each object is modeled with 3D boxes or 3D cylinders, which allow us to extend the use of our previous visibility analysis in urban environments presented for static objects [8].

2.1 Moving Car

2.1.1. 3D Modeling

As we mentioned earlier, web-cameras in urban environments can record the moving cars at any specific time. Image sources such as web cameras, like other similar sensors sources, demand an additional stage of Automatic Target Detection (ATD) algorithms to extract these objects from the image [18]. In this research we do not focus on ATD, which must be implemented when shifting from the research described in the paper toward an applicable system.

The common car structure can be easily modeled by two 3D boxes, as can be seen in Fig. 1(b), which is similar to the original car structure presented in Fig 1(a).

Fig. 1. Car Modeling Using 3D Boxes: (a) the Original Car, (b) the Modeled Car

We define the Car Boundary Points (CBP) as the set of visible surfaces' boundary points of 3D boxes modeling the car presented in Fig 1(b). Each box is modeled as 3D cubic $C_{car}(x, y, z)$ as presented extensively in [8] for a building model case:

$$C_{car}(x,y,z) = \begin{pmatrix} x = t \\ y = \begin{pmatrix} x^n - 1 \\ 1 - x^n \end{pmatrix} \\ z = c \end{pmatrix} \tag{1}$$

$$-1 \le t \le 1$$
$$n = 350$$
$$c = c + 1$$

Car Boundary Points (CBP) - we define CBP of the object i as a set of boundary points $j = 1..N_{CBP\_bound}$ of the visible surfaces of the car object, from viewpoint $V(x_0, y_{0,}z_0)$, where the maximum surface's number is six and each surface defined by four points, $N_{CBP\_bound} \le 24$.

In Fig. 2, car is modeled by using two 3D boxes. Visible surfaces colored in red, CBP marked with yellow points.

$$CBP_{i=1..N_{CBP\_bound}}(x_0, y_0, z_0) = \begin{bmatrix} x_1, y_1, z_1 \\ x_2, y_2, z_2 \\ .. \\ x_{N_{CBP\_bound}}, y_{N_{CBP\_bound}}, z_{N_{CBP\_bound}} \end{bmatrix} \tag{2}$$

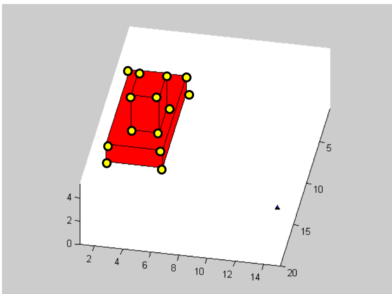


Fig. 2. Modeling Car Using 3D Boxes (CBP Marked with Yellow Points)

2.1.2. Probabilistic Visibility Analysis

Visibility has been treated as Boolean values. Due to incomplete information and the uncertainties of predicting the car's location at future times, visibility becomes much more complicated.

As it is well known from basic kinematics, CBP can be estimated in future time $t + \Delta t$ as:

$$CBP_i(t + \Delta t) = CBP_i(t) + V(t)\Delta t + \frac{A(t)\Delta t^2}{2} \tag{3}$$

Where $V(t)$ is the car velocity vector $V(t) = (v_x v_y)^T$, and the acceleration vector $A(t) = (a_x a_y)^T$. Estimation of a car's location in the future based on a web camera is not a simple task. Driver behavior generates multi-decision modeling, such as car-following behavior, gap acceptance behavior, or lane-change cases including traffic flow, speed etc.[1].

Our probabilistic car model is based on microscopic simulation models that were properly calibrated and validated using VISSIM simulation. VISSIM is a time-based microscopic simulation tool that uses various driver behaviors and vehicle performances to accurately represent an urban traffic model. The VISSIM simulation model has been validated when compared to the data from various real-world situations [4] and used for the test-bed network by [14],[16], and on driver behavior research defining average speed and acceleration [1].

VISSIM simulation environment is a very robust one, characterizing each driver/vehicle unit include: behavior of driver/vehicle unit (desired speed, desired acceleration and deceleration, sensitivity thresholds and parameters that determine behavior); technical specifications of the vehicle (e.g. length, weight, engine power, maximum speed, acceleration and deceleration potential); relative positions of preceding and following vehicles on different lanes, and the positions of objects.

The average speed in urban environments is about 45 [km/hr], from a minimum of 40 [km/hr] up to a maximum of 50 [km/hr]. In the situation of a free driving case, which is the common mode in urban environments [21], the acceleration of family car can change between $1 - 3.5 \left[{}^{m}/_{\sec^2}\right]$, and on average $2.5 \left[{}^{m}/_{\sec^2}\right]$, as seen in Fig. 3.

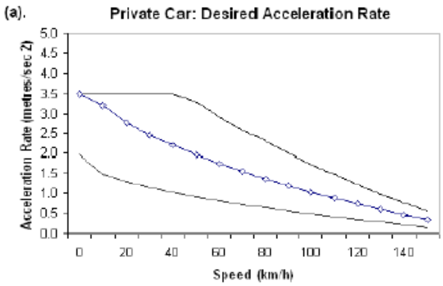


Fig. 3. Average Acceleration Rate of a Family Car in an Urban Environment [1]

As can be seen from several validations of car and driver estimation, velocity and acceleration are distributed as normal ones, and lead to normal location distribution:

$$\begin{gathered} V(t) \sim N(\mu = 45, \sigma^2 = 10) \\ A(t) \sim N(\mu = 2.5, \sigma^2 = 1) \\ CBP(t + \Delta t) \sim \sum N \end{gathered} \tag{4}$$

In time step t, where the car's location is taken from a web-camera, visibility analysis from $CBP(t)$is an exact one, based on our previous visibility analysis [8], as seen in Fig. 2. Visibility analysis becomes probabilistic for future time $t + \Delta t$, applying the same visibility analysis for $CBP(t + \Delta t)$ presented in Fig 4:

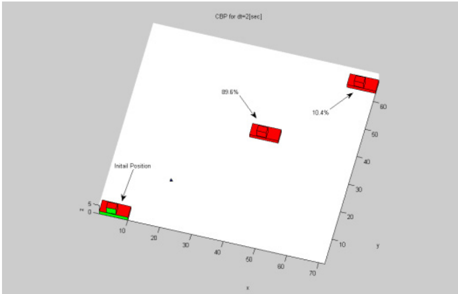


Fig. 4. Probabilistic Visibility Analysis for CBP: 3D View of Car Modeling Location from Web-oriented Source with Estimated Probabilistic Location; CBP is the Boundary for Visible Surfaces (Colored in Red)

In Fig. 4, the car's location from a web-camera appears in the bottom left side. For $\Delta t = 2[sec]$, the car's location is marked by two 3D boxes, where CBP for each of them is the boundary of visible surfaces marked in red. The probability that the visible surfaces, which are bounded by CBP, will be visible in future time is based on the last update taken from the web application (depicted with arrows in Fig. 4), computed by using two different random normal PDF values for V and A based on eq. (4).

2.2 Pedestrians

2.2.1. 3D Modeling

Pedestrian modeling can be done in high resolution, but due to Auto Target Detection (ATD) algorithms capabilities, pedestrians are usually bounded by a 3D cylinder and not as an exact detailed model [18]. For this reason, we model pedestrians as 3D cylinders, which is somewhat conservative but still applicable.

Pedestrian can be easily modeled by 3D cylinder, as seen in Fig. 5 (marked in red), which is similar to the output from ATD methods tested on a web-camera output recognizing walkers in urban environments.

We extend our previous visibility analysis concept [8] and include new objects modeled as cylinders as continuous curves parameterization, $C_{Peds}(x, y, z)$. Cylinder parameterization can be described as:

$$C_{Peds}(x, y, z) = \begin{pmatrix} r\sin(\theta) \\ r\cos(\theta) \\ c \end{pmatrix} \tag{5}$$

$$0 \le \theta \le 2\pi$$
$$c = c + 1$$
$$0 \le c \le h_{peds\_\max}$$

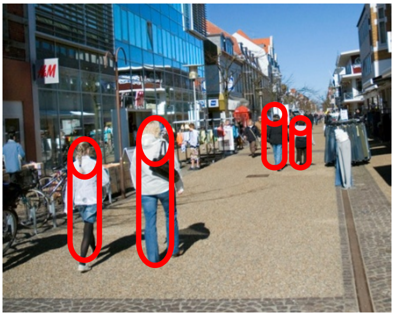


Fig. 5. Modeling Pedestrians in Urban Scene Using Cylinders (Colored in Red)

We define the visibility problem in a 3D environment for more complex objects as:

$$C'(x, y)_{z_{const}} \times (C(x, y)_{z_{const}} - V(x_0, y_0, z_0)) = 0 \tag{6}$$

where 3D model parameterization is $C(\mathrm{x,y})_{z=const}$, and the viewpoint is given as $V(x_0, y_{0,} z_0)$. Extending the 3D cubic parameterization, we also consider the cylinder case. Integrating eq. (5) to (6) yields:

$$\begin{pmatrix} r\cos\theta \\ -r\sin\theta \\ 0 \end{pmatrix} \times \begin{pmatrix} r\sin\theta - V_x \\ r\cos\theta - V_y \\ c - V_z \end{pmatrix} = 0 \tag{7}$$

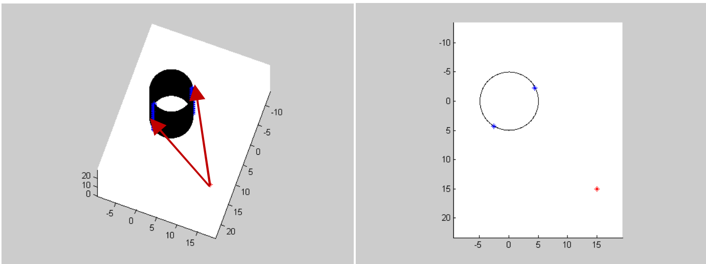
$$\theta = \arctan\left(-\frac{-r - \frac{\left(-vy\, r + \sqrt{vx^4 - vx^2 r^2 + vy^2 vx^2}\right) vy}{vx^2 + vy^2}}{vx}, -\frac{-vy\, r + \sqrt{vx^4 - vx^2 r^2 + vy^2 vx^2}}{vx^2 + vy^2} \right) \tag{8}$$

As can be noted, these equations are not related to Z axis, and the visibility boundary points are the same for each x-y cylinder profile.

The visibility statement leads to complex equation, which does not appear to be a simple computational task. This equation can be efficiently solved by finding where the equation changes its sign and crosses zero value; we used analytic solution to speed up computation time and to avoid numeric approximations. We generate two values of θ generating two silhouette points in a very short time computation. Based on an analytic solution to the cylinder case, a fast and exact analytic solution can be found for the visibility problem from a viewpoint.

We define the solution presented in eq. (8) as x-y-z coordinates values for the cylinder case as **Pedestrian Boundary Points (PBP)**. PBP are the set of visible silhouette points for a 3D cylinder modeling the pedestrian, as presented in Fig 6:

$$PBP_{i=1..N_{PBP\_bound}=2}(x_0, y_0, z_0) = \begin{bmatrix} x_1, y_1, z_1 \\ x_{N_{PBP\_bound}}, y_{N_{PBP\_bound}}, z_{N_{PBP\_bound}} \end{bmatrix} \tag{9}$$



(a) (b)

Fig. 6. Pedestrian Boundary Points (PBP) for a Cylinder using Analytic Solution marked as blue points, Viewpoint Marked in Red: (a) 3D View (Visible Boundaries Marked with Red Arrows); (b) Topside View

2.2.2. Probabilistic Visibility Analysis

The kinematic model of a pedestrian is only a part of the estimation and prediction of his movement in an urban environment. For simplicity, we use only a kinematic model for pedestrian's future location, since decision-making in this field is very complicated.

A well-known and common kinematic model for pedestrians is presented by Hoogendoorn et al. [9]. Based on this model, pedestrian location can be predicted as:

$$PBP(t + \Delta t) = PBP(t) + V(t)\Delta t + w \tag{10}$$

where w is a white noise of a standard Wiener Process which reflects the uncertainty in the expected traffic condition, described as Gaussian distribution.
Pedestrian speed V can be divided into three major groups:

1. Fast: 1.8 meter per second
2. Standard: 1.3 meter per second
3. Slow: 0.85 meter per second

$$\begin{gathered} V(t) \sim N(\mu = 1.3, \sigma^2 = 0.5) \\ w \sim \sqrt{\Delta t} N(0,1) \\ PBP(t + \Delta t) \sim \sum (N + \sqrt{\Delta t} N) \end{gathered} \tag{11}$$

In time step t, where pedestrian location is taken from a web-camera, visibility analysis from $PBP(t)$ can be computed.

Visibility analysis becomes probabilistic for a future time $t + \Delta t$. Applying the same visibility analysis for $PBP(t + \Delta t)$ and basing ourselves on the probabilistic model, leads to a probabilistic visibility definition. In Fig. 7, pedestrian location from a web-camera is marked as a black cylinder, where PBP are marked with blue points. For $\Delta t = 3[sec]$, pedestrian location is marked with green cylinders, where PBP for each pedestrian is marked in black. The probability that the surface bounded by PBP will be visible in future time is presented from the last update taken from the web application, presented with arrows in Fig. 7, using four different random normal PDF values for V and w based on eq. (11).

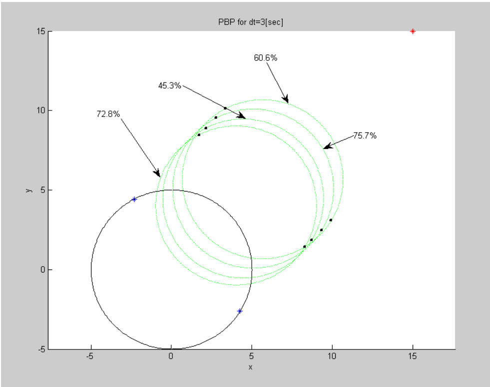


Fig. 7. Probabilistic Visibility Analysis for PBP: Topside View of Pedestrian Location from Web-oriented Source Marked with Black Cylinder, Estimated Probabilistic Location Marked in Green, PBP Marked with Black Points

2.3 Trees

2.3.1. 3D Modeling

Unlike the two previous objects, trees do not move. However, tree branches, which are very common in urban environments, are greatly affected by wind.

Modeling and analyzing the capability to 'see' through trees is not a trivial undertaking, and is related to many factors such as tree type, age, number and shape of leaves, wind profile etc.

We model tree structure as two cylinders, one for the tree stem and a second, larger one, above the first for the upper part of the tree, as seen in Fig 8.

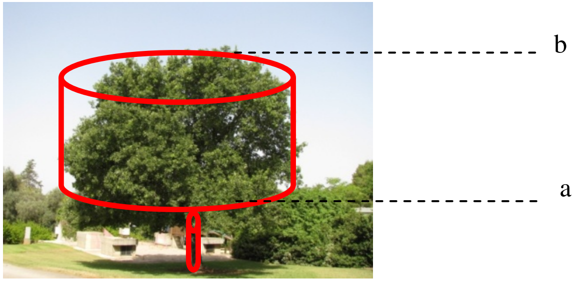
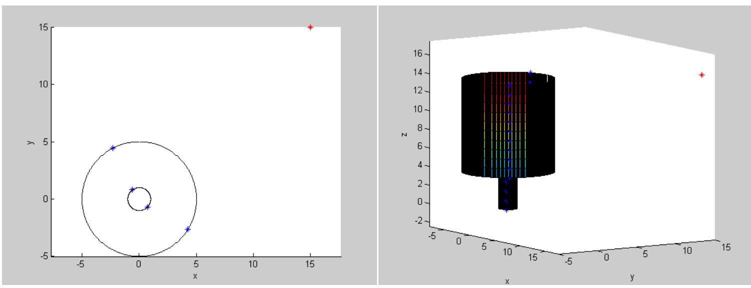


Fig. 8. Modeling Tree by Two Cylinders colored in Red: Minimum and Maximum Higher Cylinder Boundaries [a, b] marked with Dotted Lines

Tree branches are taken into account by the probabilistic model and wind effect, as detailed in the next sub-section.

We define the **Tree Boundary Points (TBP)** as the set of visible silhouette points for two 3D cylinders, modeling as presented in Fig 9, with TBP marked as blue points (TBP is also the solution to eq. (6) for two cases of cylinder parameterization):

$$TBP_{i=1..N_{TBP\_bound}=4}(x_0, y_0, z_0) = \begin{bmatrix} x_1, y_1, z_1 \\ x_2, y_2, z_2 \\ \ldots \\ x_{N_{NBP\_bound}}, y_{N_{NBP\_bound}}, z_{N_{NBP\_bound}} \end{bmatrix} \tag{12}$$



(a) (b)

Fig. 9. TBP for Tree Object Modeled by Two Cylinders colored in Black: (a) Topside View (TBP Marked in Blue Point); (b) 3D View

2.3.2. Probabilistic Visibility Analysis

In the case of the tree, we focus on branches' movements in the presence of a wind. Modeling a tree as a dynamic system is a very complex task; such a system is affected by an unstable phenomenon – the wind.

We use Finite Element (FE) analysis solution to predict tree motion in the presence of the common wind profile introduced by [10].

As for natural wind profile, wind speed $v_w(z)$ increases as the distance from the ground increases. In our simulation, wind profile is approximated according to the following logarithmic law:

$$v_w(z) = \ln\left[1 + z(e-1)/(b-a)\right] \tag{13}$$

The outcome of the wind profile - described in eq. (13) – and its effect on tree branches' displacement can be simulated as sinusoidal function with a frequency of f=0.27 Hz (y-axis displacements can be neglected relative to x-axis displacements) [10]. Based on validated models [10], tree branches' movements in the presence of wind can be simulated as:

$$TBP(t+\Delta t)_{i=3}^{4} \sim TBP(t)_{i=3}^{4} \cdot \sin(f(t+\Delta t)) \cdot U(a,b) \tag{14}$$

where a and b are the minimum and the maximum highs of the second 3D cylinder presented in Fig.8.

It should be mentioned that $TBP(t+\Delta t)_{i=1}^{2}$ (the first 3D cylinder modeling tree stem) is a static object which is rarely affected by the wind. The probabilistic tree modeling in time,$TBP(t+\Delta t)_{i=3}^{4}$ leads directly to a probabilistic visibility solution.

In Fig. 10, tree modeling from a web-camera is marked as a black cylinder, where TBP are marked with blue points. For $\Delta t = 15[sec]$, in the presence of wind, estimation of tree branches movement is marked with green cylinders, where TBP for each of them is marked in red. The probability that the surface, which is bounded by TBP, will be visible at a future time is based on the last update taken from a web application (as presented with arrows in Fig. 10), and is computed by using TBP future time prediction, presented in eq. (14). It can be noticed that $TBP(t+\Delta t)_{i=1}^{2}$ stays constant.

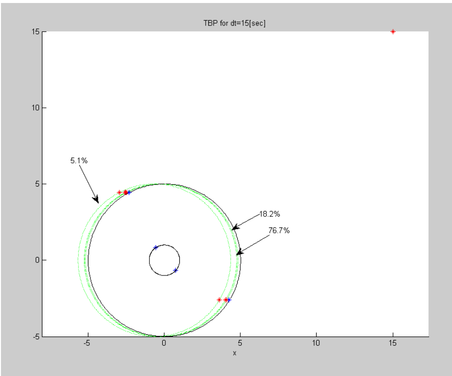


Fig. 10. Probabilistic Visibility Analysis for TBP: Topside View of the Tree Modeling with Probabilistic Visibility Analysis for Tree Branches in the Presence of Wind (Estimated Probabilistic Location Marked in Green Cylinders, TBP Marked in Red Points)

3 Problem Statement

We consider visibility problem in a 3D urban environment, consisting of static constant objects and dynamic objects.

Given:

- Static objects:
  - 3D buildings modeled as 3D cubic parameterization $\sum_{i=1}^{N_{of\_build}} C_i(x, y, z =_{h_{\min}}^{h_{\max}})$
- Dynamic objects:
  - Moving cars modeled as 3D cubic parameterization, $C_{car}(x, y, z)$
  - Pedestrian modeled as cylinder parameterization , $C_{peds}(x, y, z)$
  - Trees modeled with two cylinder parameterization, $C_{tree}(x, y, z)$
- Wind profile $v_w(z)$ and Viewpoint $V(x_0, y_{0,} z_0)$, in 3D coordinates.

Computes:

Set of all visible points in $\sum_{i=1}^{N}[C_{building_i}, C_{car_i}, C_{tree_i}, C_{peds_i}]$ from $V(x_0, y_{0,} z_0)$,.

We extend our previous work [8], developed for a fast and efficient visibility analysis for buildings in urban environments, and consider also a basic structure of cylinders, which allows us to model pedestrians and trees. Based on our probabilistic visibility computation of dynamic objects, we test the effect of these by using data gathered from web-oriented GIS sources to update our estimation and prediction on these entities.

4 Simulations

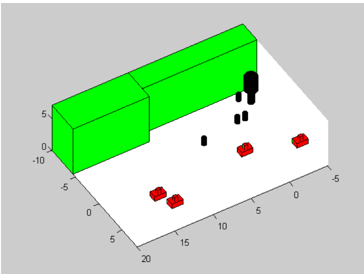
We demonstrate dynamic objects' effect on visibility analysis in a 3D urban environment using source data from web cameras located at constant points.

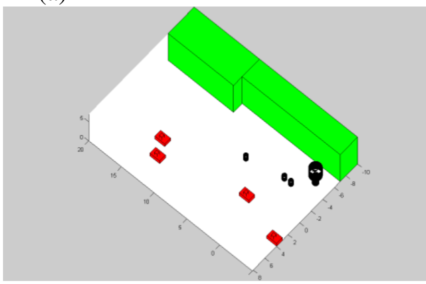
Based on the availability of web cameras on the internet, we approximated the images collected from the web cameras, and tested and analyzed the environment using 3D box and cylinder modeling. The tests were carried out on a 1.8GHz Intel Core CPU and the algorithms were developed with Matlab.

We used a web-camera data source located at 14th St, East Village NYC, U.S. [available at `http://joemaller.com/webcam/`]. The map showing the approximate area visible from the webcam is presented in Fig. 11. The Web camera image with moving cars, pedestrians and trees is presented in Fig. 12(a), and the approximated modeling of these features by using 3D boxes and cylinders can be seen in Fig 12 (b),(c).

Fig. 11. Approximate Area Visible from the Webcam Located at 14th St, East Village NYC, U.S. [available at `http://joemaller.com/webcam/`]

(a)





(b) (c)

Fig. 12. A Real Webcam-based Scene: (a) 14th St, East Village NYC Webcam Image with Moving Cars, Pedestrians and Trees; (b) Approximated 3D Model using 3D Boxes and Cylinders (Moving Cars Colored in Red, Buildings in Green, Pedestrians and Tree in Black); (c) Approximated 3D model from other 3D direction

Prediction from the current image of the web camera is the basic data source. Based on that, we estimate and predict the dynamic object's location at a future time, as described in Fig. 13(a) and Fig. 13(b).

In Fig. 13(c) we present the probabilistic visibility analysis where dynamic objects affect the visibility from the viewpoint to the buildings. We predict the dynamic objects' location for a future time, t+Δt, where $\Delta t = 1[sec]$. The visible and hidden parts of the surfaces of the constant object, i.e. in this case, buildings, are determined by values between [0,1]. The probability that the dynamic object will be located at a specific point in a future time sets the visibility value, based on the web-camera image generated in the past. In Fig. 13(d), we present the same analysis from other viewpoint.

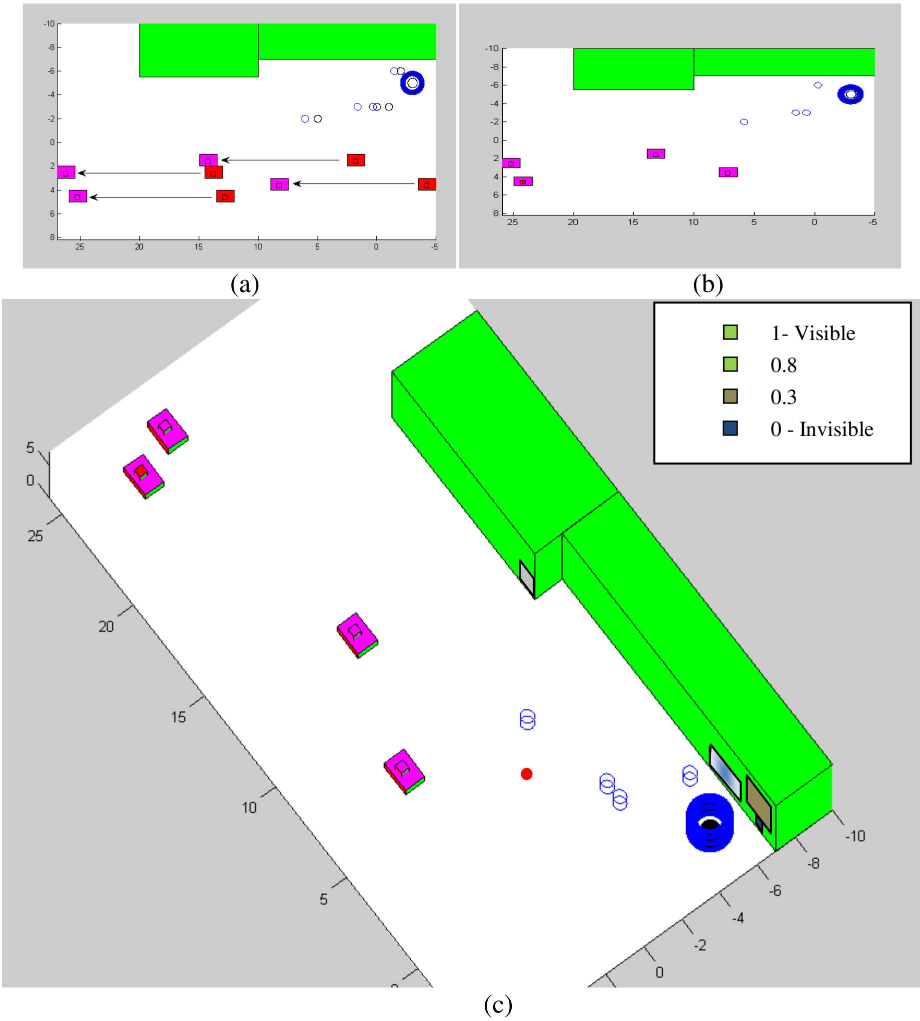
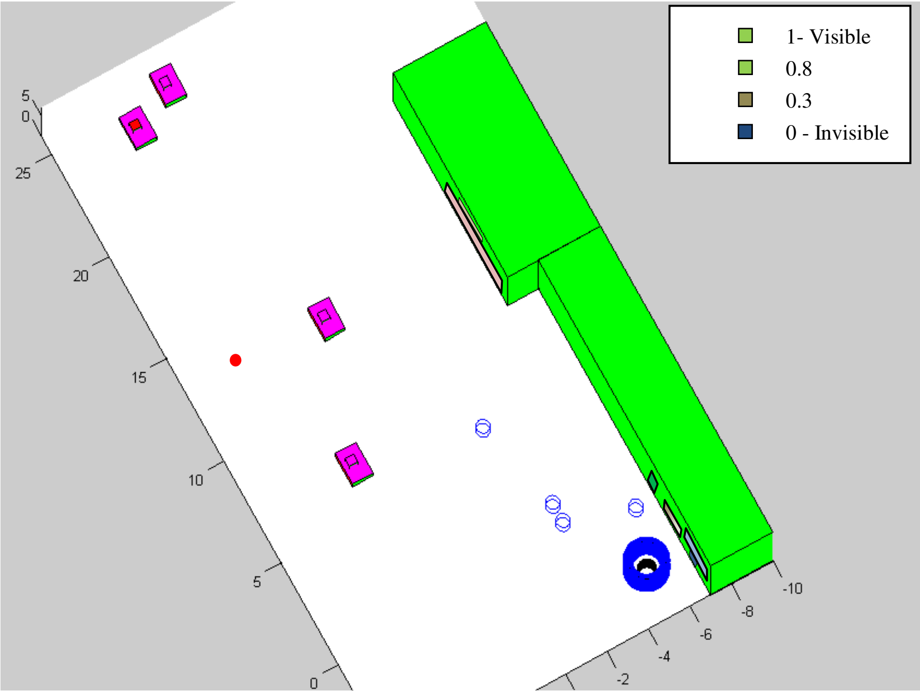


Fig. 13. Visibility Analysis of Scene no. 1: (a) Image from the Webcam at Time *t* with Estimated Car Location Marked with Arrows; (b) Scene in a Future Time; (c) Probabilistic Visibility Analysis, Viewpoint Marked in Red, Visible and Invisible Parts Marked in Different Colors on the Static Objects; (d) Probabilistic Visibility Analysis from another Viewpoint marked in red.



(d)

Fig. 13. (*continued*)

4.1 3D Model Construction From 2D Image

Modeling dynamic objects by using web-based data, such as cameras, is not a simple task. Creating a 3D scene from a 2D image is becoming more and more viable due to advanced hardware and image-processing computer vision algorithms [12]. Saxena et al. [19], presented a 3D scene structure algorithm from a single image using Markov Random Field (MRF) method. The algorithm was extensively tested and successfully applied on large-scale 3D models.

Nowadays, several products can be used to build a 3D model in urban scenes from a 2D image, such as [12],[13],[20].

Our simulations are based on the assumption that the images which are taken from web servers, such as Google Map web cameras, are reconstructed to 3D models using the current product and algorithms in this field. For simplicity, we approximated the dynamic and static objects from the scenes, as can be seen in the following simulations.

4.2 Efficiency

Efficient LOS-based visibility methods for DEM models, such as Xdraw, have been introduced in order to generate approximate solutions [6]. However, the computation time of these methods is at least $O(n(n-1)$, and, above all, the solution is an approximate one. Complexity analysis for simple boxes is detailed in [8]. Extending a visibility algorithm, while also considering 3D cylinders with an analytic solution for visibility boundaries, is meaningless from the complexity aspect.

5 Conclusions and Future Work

We have presented an efficient algorithm for visibility computation in urban environments, which considers dynamic objects integrated to the scene. The urban environment is divided into static objects such as basic buildings, and to dynamic objects like moving cars, pedestrians and trees.

Due to the inherent character of dynamic objects, we must predict and estimate the future location of these objects in the environment. We present a probabilistic visibility analysis which is related to the probability of an object's presence in the environment and the effect on visibility analysis from validated models of transporting and human behavior fields. To illustrate our concept, we used web-cameras located at constant points in the environments, in order to update our model at each time period (as is already being done in the Google Maps application).

Moving car and basic building structures are modeled with mathematical approximating, for presentation of buildings' corners and car boundary modeling, by using 3D boxes. Pedestrians and trees are modeled by cylinders parameterization and extending the basic model structure. We formulated our visibility algorithm based on a fast visibility boundary challenging fast and exact hidden surfaces computation.

We tested dynamic objects' effect on urban environments by using web-cameras updating dynamic objects' location at certain time periods in the 3D model, as can be seen in simulations. The efficiency and complexity which has been presented in our previous work is still valid for 3D cylinders with probabilistic visibility.

The main contribution of the presented method in this paper is challenging dynamic objects' effect on visibility aspects in urban environments using a probabilistic concept, defining and formulating cylinder entity using fast visibility computation, all this without the need for special hardware.

Further research will focus on modeling and exploring other tree profiles with methods other than cylinders, and also consider other dynamic entities such as bicycle riders and non-rectangular building shapes.

References

1. Archer, J.: Methods for the Assessment and Prediction of Traffic Safety at Urban Intersections and their Application in Micro-simulation Modeling, Centre for Traffic Simulation Research, CTR, Sweden. Technical Report (2010)
2. Doytsher, Y., Shmutter, B.: Digital Elevation Model of Dead Ground. In: Symposium on Mapping and Geographic Information Systems (Commission IV of the International Society for Photogrammetry and Remote Sensing), Athens, Georgia, USA (1994)
3. Durand, F.: 3D Visibility: Analytical Study and Applications. PhD thesis, Universite Joseph Fourier, Grenoble, France (1999)

4. Fellendorf, M., Vortisch, P.: Validation of the Microscopic Traffic Flow Model VISSIM in Different Real world Situations. In: 79th Annual Meeting of Transportation Research Board, UK (2001)
5. De Floriani, L., Magillo, P.: Visibility Algorithms on Triangulated Terrain Models. International Journal of Geographic Information Systems 8, 13–41 (1994)
6. Franklin, W.R., Ray, C.: Higher isn't Necessarily Better: Visibility Algorithms and Experiments. In: Waugh, T.C., Healey, R.G. (eds.) Advances in GIS Research: Sixth International Symposium on Spatial Data Handling, pp. 751–770. Taylor & Francis, Edinburgh (1994)
7. Franklin, W.R.: Siting Observers on Terrain. In: Proc. of 10th International Symposium on Spatial Data Handling, pp. 109–120. Springer (2002)
8. Gal, O., Doytsher, Y.: Fast and Accurate Visibility Computation in a 3D Urban Environment. In: Proc. of the Fourth International Conference on Advanced Geographic Information Systems, Applications, and Services, Valencia, pp. 105–110 (2012)
9. Hoogendoorn, S.P., Bovy, P.H.L.: Pedestrian route-choice and activity scheduling theory and models. Transportation Research Part B 38, 169–190 (2004)
10. Hu, X., Tao, W., Guo, Y.: Using FEM to predict tree motion in a wind field. Journal of Zhejiang University, Science A 9(7), 907–915 (2008)
11. Nadler, B., Fibich, G., Lev-Yehudi, S., Cohen-Or, D.: A Qualitative and Quantitative Visibility Analysis in Urban Scenes. Computers & Graphics 5, 655–666 (1999)
12. Make3D, `http://make3d.cs.cornell.edu`
13. PictoGraph, `http://palentier.blogsopt.co.il/2011/11/3d-prehistoric-pictograph-printing.html`
14. Park, B., Schneeberger, J.D.: Microscopic Simulation Model Calibration and Validation: Case Study of VISSIM Simulation Model for a Coordinated Actuated Signal System, Transportation Research Record 1856 , Paper No. 03-2531
15. Plantinga, H., Dyer, R.: Visibility, Occlusion, and Aspect Graph. The International Journal of Computer Vision 5, 137–160 (1990)
16. Parker, D., Lajunen, T.: Are Aggressive People Aggressive Drivers? A Study of the Relationship between Self-Reported General Aggressiveness Driver Anger and Aggressive Driving. Accident Analysis and Prevention 33(2), 243–255 (2001)
17. Ratti, C.: The Lineage of Line: Space Syntax Parameters from the Analysis of Ur-ban DEMs. Environment and Planning B: Planning and Design 32, 547–566 (2005)
18. Song, Y.: The research of a new Auto Target Recognition directed Image compression. In: 3rd Int. Congress on Image and Signal Processing (CISP), China, October 16-18 (2010)
19. Saxena, A., Sun, M., Ng, A.Y.: Make3D: Learning 3D Scene Structure from a Single Still Image. IEEE Transaction on Pattern Analysis and Machine Intelligence, PAMI (2008)
20. 3D Studio Max, `http://usa.autodesk.com/3ds-max`
21. Wiedemann, R., Reiter, U.: Microscopic Traffic Simulation: The Simulation System MISSION, Background and Actual State. Project ICARUS (V1052), Final Report, Brussels CEC.2: Appendix A (1992)
22. Wang, J., Robinson, G.J., White, K.: A Fast Solution to Local Viewshed Computation Using Grid-based Digital Elevation Models. Photogrammetric Engineering & Remote Sensing 62, 1157–1164 (1996)
23. Wang, J., Robinson, G.J., White, K.: Generating Viewsheds without Using Sightlines. Photogrammetric Engineering & Remote Sensing 66, 87–90 (2000)
24. Zlatanova, S., Rahman, A., Wenzhong, S.: Topology for 3D Spatial Objects. In: International Symposium and Exhibition on Geoinformation, pp. 22–24 (2002)